```
# UCSC Extension
# DBDA.X409.(10) MySQL and Oracle Database for Developers and
Designers
#
# Final Test
# Part 3
# Cursor
#
# Cheng Fei
# 11/17/2022
#
# 1. The procedure identifies all the tables that are present in the
database.
# 2. The procedure then iterates through the cursor to create a
backup table for every table present in the database.

# The following is the expected output after you run the procedure.
# Example: For STUDENT table there should be an equivalent
STUDENT_BKP table created.
# Similarly for all tables there should be an equivalent
<TABLENAME_BKP> table created.

# Change the current database to studentdb.
USE studentdb;

# @_CREATE_PROCEDURE_
# A database procedure using dynamic SQL.
# Create a backup for each table in the studentdb database.
DELIMITER $
DROP PROCEDURE IF EXISTS backup_all_tables$
CREATE PROCEDURE backup_all_tables()
BEGIN
        DECLARE v_table_name VARCHAR(100);
    DECLARE done INT DEFAULT 0;
    DECLARE v_sql VARCHAR(1000);
    DECLARE table_cur CURSOR FOR
                SELECT table_name
        FROM information_schema.tables
        WHERE table_schema = 'studentdb';
        BEGIN
                DECLARE EXIT HANDLER FOR NOT FOUND SET done = 1;
        OPEN table_cur;
                        table_loop: LOOP
                                FETCH table_cur INTO v_table_name;
                IF (done = 1) THEN
                                        LEAVE table_loop;
                END IF;
                SET @v_sql := concat('DROP TABLE IF EXISTS ',
v_table_name, '_bkp');
                PREPARE stmt FROM @v_sql;
                EXECUTE stmt;
                SET @v_sql := concat('CREATE TABLE ', v_table_name,
'_bkp ',
```

```
         'AS ',
                                        'SELECT * FROM ',
v_table_name);
                                   SELECT @v_sql script;
                PREPARE stmt FROM @v_sql;
                EXECUTE stmt;
            END LOOP table_loop;
        CLOSE table_cur;
    END;
END$
DELIMITER ;

# Commit changes.
COMMIT;
```